

AEC: Web Development

Steps to run React JS applications

- 1: Install Node.js : Download and install Node.js from the official website. This installs Node, npm, and npx which are required to run React applications.
- 2: Create a React Application : `Open Command Prompt (CMD)` from `the VS Code` and type `npx create-react-app calculator`. This command creates a new React project named calculator.
- 3: Navigate to the Project Folder: `cd calculator`
- 4: Open the Project in VS Code: `code .` (code<space>dot)
- 5: Replace the Content of src/App.js with the respective code of the program and save it.
- 6: Run the Application: `npm start`

Execution steps for programs 1 to 3 remain same.

1. Calculator Application

```
import { useState } from "react";
function App() {
  const [a,setA]=useState(0);
  const [b,setB]=useState(0);
  const [r,setR]=useState(0);
  return (
    <div>
      <h2>Calculator</h2>
      <input type="number" onChange={(e)=>setA(+e.target.value)} />
      <input type="number" onChange={(e)=>setB(+e.target.value)} />
      <br/><br/>
      <button onClick={()=>setR(a+b)}>+</button>
      <button onClick={()=>setR(a-b)}>-</button>
      <button onClick={()=>setR(a*b)}>*</button>
      <button onClick={()=>setR(a/b)}>/</button>
      <h3>Result: {r}</h3>
    </div>
  );
}
export default App;
```

2. React Voting Application

```
import {useState} from "react";
function App() {
  const [a,setA]=useState(0);
  const [b,setB]=useState(0);
  return (
    <div>
      <h2>Voting App</h2>
      <h3>Candidate A: {a}</h3>
      <button onClick={()=>setA(a+1)}>Vote A</button>
      <h3>Candidate B: {b}</h3>
      <button onClick={()=>setB(b+1)}>Vote B</button>
    </div>
  );
}
export default App;
```

3. React Login Interface with Validation

```
import {useState} from "react";
function App() {
  const [user,setUser]=useState("");
  const [pass,setPass]=useState("");
  const [msg,setMsg]=useState("");

  const login=()=>{
    if(user==="" || pass==="")
      setMsg("Enter Username and Password");
    else
      setMsg("Login Successful");
  }

  return (
    <div>
      <h2>Login</h2>
      <input placeholder="Username" onChange={(e)=>setUser(e.target.value)} />
      <br/><br/>
      <input type="password" placeholder="Password" onChange={(e)=>setPass(e.target.value)} />
      <br/><br/>
      <button onClick={login}>Login</button>
      <p>{msg}</p>
    </div>
  );
}
export default App;
```

4. React Bootstrap Form

After navigating to the said folder(step 3),

Install Bootstrap : `npm install react-bootstrap bootstrap`

Then Modify src/App.js as below

```
import 'bootstrap/dist/css/bootstrap.min.css';
import {Form,Button} from "react-bootstrap";
function App(){
  return(
    <div className="container">
      <h2>User Form</h2>
      <Form>
        <Form.Group>
          <Form.Label>Name</Form.Label>
          <Form.Control type="text" />
        </Form.Group>
        <Form.Group>
          <Form.Label>Email</Form.Label>
          <Form.Control type="email" />
        </Form.Group>
        <Form.Group>
          <Form.Label>Age</Form.Label>
          <Form.Control type="number" />
        </Form.Group>
        <br/>
        <Button>Submit</Button>
      </Form>
    </div>
  );
}
export default App;
```

5. Responsive Image Interaction

Execution steps same as program 4.

```
import 'bootstrap/dist/css/bootstrap.min.css';
function App(){
  return(
    <div className="container text-center">
      <h2>Responsive Image</h2>
      alert("Image Clicked")}
      />
    </div>
  );
}

export default App;
```

6. Node.js Custom Server (HTTP, OS, Path, Event)

Execution steps as below.

- 1: Create Folder : `mkdir node-server`
- 2: Enter Folder: `cd node-server`
- 3: Create File `server.js`
- 4: Add Code to the file.
- 5: Run Server: `node server.js`
- 6: Open Browser: `http://localhost:3000`

```
const http=require("http");
const os=require("os");
const path=require("path");
const EventEmitter=require("events");
const event=new EventEmitter();
event.on("msg",()=>{ console.log("Event Triggered"); });

http.createServer((req,res)=>{
  event.emit("msg");
  res.write("OS:"+os.platform()+"\n");
  res.write("File:"+path.basename(__filename));
  res.end();
}).listen(3000);

console.log("Server running at port 3000");
```

7. Express JS Web Server

Execution steps as below.

1: Create Folder: `mkdir express-app`

2: Enter Folder: `cd express-app`

3: Initialize Node : `npm init -y`

4: Install Express : `npm install express`

5: Create File `app.js`

6: Add Code

7. Run Server : `node app.js`

8: Open Browser : `http://localhost:3000`

```
const express=require("express");
const app=express();
app.get("/",(req,res)=>{ res.send("Hello from Express Server"); });

app.listen(3000,()=>{ console.log("Server running on port 3000"); });
```

8. MongoDB Basic Operation

1. Install MongoDB Community Server.

2. Create project folder:

```
mkdir mongo-app  
cd mongo-app
```

3. Initialize project.

```
npm init -y
```

4. Install MongoDB driver.

```
npm install mongodb
```

5. Create file `db.js`.

6. Add Code

7. Run program.

```
node db.js
```

```
const {MongoClient}=require("mongodb");  
MongoClient.connect("mongodb://127.0.0.1:27017")  
.then(client=>{  
  const db=client.db("test");  
  db.collection("users").insertOne({  
    name:"John",  
    age:22  
  });  
  console.log("Data Inserted");  
});
```


9. MERN Stack Full Application

Backend Setup

Create folder>mkdir backend>cd backend

Install packages

npm init -y

npm install express mongoose cors

Create server.js

```
const express=require("express");
const mongoose=require("mongoose");
const cors=require("cors");

mongoose.connect("mongodb://127.0.0.1:27017/mern");

const User=mongoose.model("User",{name:String});

const app=express();
app.use(cors());
app.use(express.json());

app.post("/add",async(req,res)=>{
  const u=new User(req.body);
  await u.save();
  res.send("Saved");
});

app.get("/users",async(req,res)=>{
  const data=await User.find();
  res.send(data);
});

app.listen(5000);
Run backend
node server.js
```

Frontend Setup

Create React project: npx create-react-app frontend>cd frontend

Install axios: npm install axios

Edit src/App.js

Run frontend

npm start

```
import {useState,useEffect} from "react";
import axios from "axios";

function App(){

  const [name,setName]=useState("");
  const [users,setUsers]=useState([]);

  const add=()=>{  axios.post("http://localhost:5000/add",{name});  }

  useEffect(()=>{
    axios.get("http://localhost:5000/users")
    .then(res=>setUsers(res.data));
  },[]);

  return(
    <div>

      <input onChange={(e)=>setName(e.target.value)} />
      <button onClick={add}>Add</button>

      {users.map(u=><p key={u._id}>{u.name}</p>)}

    </div>
  );
}

export default App;
```